Advancing Intelligent Transportation Systems: A Deep Learning Approach to Visual Object Detection and Tracking With YOLO Architectures and Deepsort

[1]Abe Degale Desta, [2]Jian Cheng

[1]School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu, China, degele@wcu.edu.et or ORCID: 0009-0007-6505-3970
[2]School of Information and Communication Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu, China, chengjia@uestc.edu.cn

ABSTRACT: The fastest advancement of deep learning has significantly enhanced multi-object detection and tracking competences, particularly in intelligent transportation systems. This article presents a comprehensive framework leveraging state-of-the-art in YOLO family architectures specifically YOLOv8 and the newly introduced YOLOv11 integrated with DeepSORT for strong multi-object tracking in smart cities though complex traffic environments. The authors evaluate our approach on public benchmark datasets including BDD100K, KITTI, and COCO, representative superior performance compared to existing methods. The proposed system realises that a mean Average Precision (mAP) of 0.914 at IoU=0.5 with YOLOv8 and more improvements with YOLOv11, while keeping real-time processing competences of 45 FPS. Our approach solves important problems including occlusions, size changes, and changes in light by using powerful feature pyramid networks, anchor-free detection, and efficient tracking relations. Combining these cutting-edge technology makes for a strong way to keep an eye on and control traffic in smart city settings.

Keywords*: Object Detection, Multi-Object Tracking, YOLOv8, YOLOv11, DeepSORT, Intelligent Transportation Systems, Deep Learning and Computer Vision.*

I. Introduction

The exponential growth of urban populations and vehicular traffic has needed the implementation of sophisticated intelligent transportation systems (ITS) like smart cities. Such kind of systems is the right capability for accurate and real-time multi-object detection and tracking (MODT), which enables applications ranging from traffic flow optimization to autonomous driving and public safety surveillance [1], [2]. Recent advancements in deep learning, particularly convolutional neural networks (CNNs), have transformed object detection, while with the You Only Look Once (YOLO) family evolving as a leading architecture due to its optimal balance between speed and accuracy [3].

The transition from YOLOv1 to the most recent YOLOv11 signifies significant advancements in detection capabilities, incorporating technologies such as feature pyramid networks (FPN), path aggregation networks (PAN), and anchor-free detection [4], [5]. These improvements have fixed problems with finding small objects and dealing with changes in scale. This makes YOLO family designs especially good for complicated traffic situations. DeepSORT has become a standard approach for monitoring multiple objects. It uses deep learning descriptors and the Simple Online and Real-time monitoring (SORT) framework to keep item IDs consistent between frames [6]. Combining advanced detection models with advanced tracking algorithms makes a powerful pipeline for fully understanding scenes in changing contexts.

This article makes a number of important contributions:

- A full review of YOLOv8 and the new YOLOv11 on a number of benchmark datasets
- The combination of advanced YOLO architectures with DeepSORT for strong multi-object tracking
- A thorough look at how well the system works in different environments, such as when objects are blocked, when the scale changes, and when the light changes
- Practical implementation insights for real-world ITS applications

II. Related Work

A. Evolution of Visual Object Detection in Deep Learning Application

The Expanding Use of Deep Learning Applications for Detecting Visual Objects. Additionally, there have been a lot of modifications in visual object detection throughout the days when features were used to the days when deep learning was used.

In the past, manually generated features like Haar-like features [7] and Histogram of Oriented Gradients (HOG) [8] were used with classifiers like Support Vector Machines (SVMs). However, when conditions changed or proved challenging, these tactics proved unsuccessful as well. Two-stage detectors of deep learning such as R-CNN [9], Fast R-CNN [10], and Faster R-CNN [11] were made possible in meeting the timely modern detection. These detectors were straightforward to set up, but their mathematical skills were insufficient. While s ingle-stage detectors that provide direct bounding box predictions, such as SSD [12] and the YOLO family [3], [13], solved the issue. This allowed for far faster inference speeds while maintaining excellent accuracy.

B. The Evolution of the YOLO Architecture

Since its inception, the YOLO architecture has seen significant development. The single-stage detection concept was first used by YOLOv1 [3], and subsequent iterations included numerous more characteristics. Anchor box mechanisms were improved by YOLOv2 [14], multi-scale detection was implemented by YOLOv3 [15], and backbone networks and training approaches were enhanced by YOLOv4 [16]. From YOLOv5 to YOLOv7, the architecture improved steadily, making it simpler to strike a balance between efficiency and performance [17].

YOLO8 is a major step forward because it has a superior backbone network, an anchor-free detecting head, and better feature fusion capabilities [4]. The new YOLOv11 builds on this by offering more ways to find little things and make the machine work faster [5].

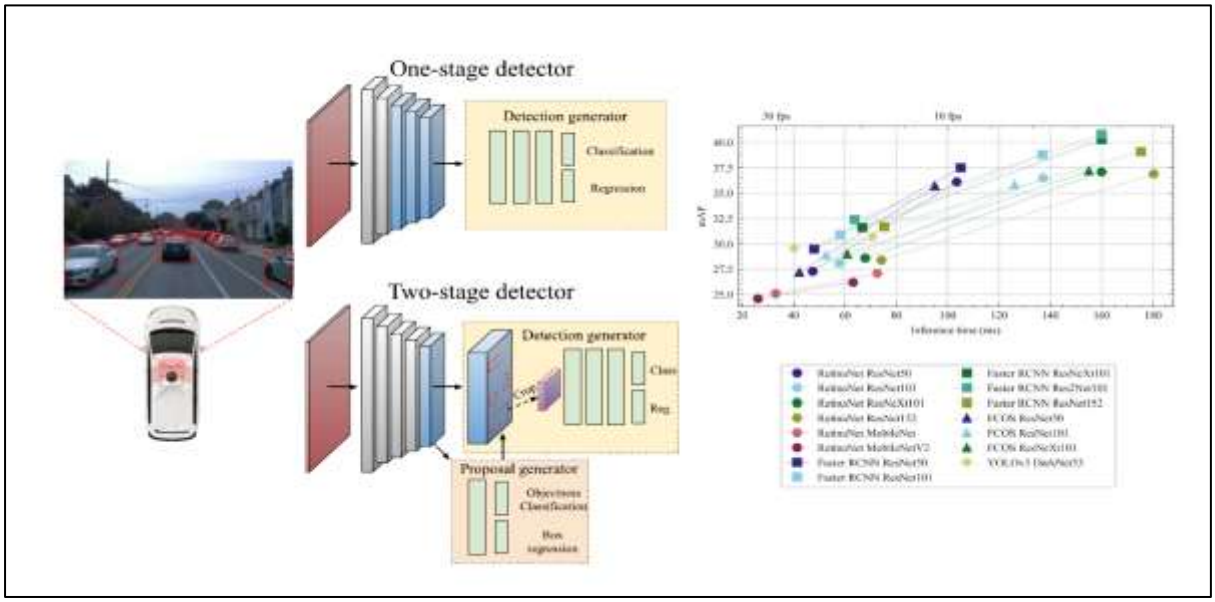
C. Algorithms for tracking multiple objects

Multi-object tracking has progressed from conventional techniques like Kalman filters [18] and particle filters [19] to contemporary deep learning-based methodologies. SORT [20] offered a straightforward yet efficient architecture, subsequently enhanced by DeepSORT [6] with deep appearance descriptors to better maintain identification during occlusions.

Recent methods have used transformer topologies [21] and graph neural networks [22] to improve tracking performance. But DeepSORT is still quite popular since it is fast and works well in real-world situations.

The architectural formation of deep learning-based object detectors is essentially defined by the trade-off between the two-stage and one-stage approaches. Two-stage detectors, such as the R-CNN family [3], operate through two stages: at first, they generate a small number of region proposals, and then they classify and regress bounding boxes on each proposal. This method is extremely accurate, but it is challenging to compute, which makes it challenging to utilize in real time.

In response, one-stage algorithms such as YOLO [2] and SSD [12] were made to do detection in one shot by right away predicting object boundaries and classifications from visual data. This unified approach does away with the proposal generation stage, which speeds up prediction by a large amount—often by an order of magnitude resulting in real-time applications possible, though not without a minor loss of accuracy, especially for small objects. A speed-versus-accuracy graph (see Fig. 1) shows the main trade-off very clearly. Two-stage methods (such as Faster R-CNN) have been in the high-accuracy area, while one-stage methods (such as YOLO and SSD) have been in the high-speed area. This boundary is always being pushed by new architectural innovations.



III. Methodology

A. System Architecture

The proposed system comprises three main components: object detection using YOLOv8 and YOLOv11, feature extraction for appearance modelling, and data association using DeepSORT. The overall pipeline processes video frames through these components to generate consistent object trajectories.

B. YOLOv8 and YOLOv11 Detection Framework

1) YOLOv8 Architecture:

YOLOv8 announced a suite of key architectural and training enhancements that significantly spread its object detection capabilities. According to the model's technical report by Jocher et al.,[25] its core

architecture is constructed upon a CSPDarknet53 backbone, which employs cross-stage partial connections to strengthen gradient flow while maintaining high computational efficiency. For feature fusion, the model integrates a Path Aggregation Network (PAN) to successfully aggregate multi-scale features, attractive the detection of objects at various sizes. A main design is the adoption of an anchor-free detection head, which simplifies the output representation by directly predicting object bounding box, thereby eliminating the difficulty and hyper parameter sensitivity associated with predefined anchor boxes. Furthermore, these architectural improvements are supported by advanced data augmentation strategies, including translation, augmentation and MixUp, which create more robust training datasets by combining multiple images, ultimately improving the model's generalization and performance Jocher et al., [25].

2) YOLOv11 Enhancements:

Based on the enhancements, the official technical report for YOLOv11 presents a number of significant improvements that together strengthen visual object detection performance. For more realistic multi-scale feature fusion, YOLOv11 features an expanded description of pyramid network with bi-directional connections (BiFPN) in addition to an active head architecture that allows scale-aware learning of features by adaptively focusing on various object sizes and frames. Optimized training policies, including self-adversarial training, which increases model robustness by testing it with adversarial manipulated data, assist these structural advances. Furthermore, the model makes use of enhanced loss functions created especially for more accurate and reliable bounding box regression, which raises localization accuracy in a variety of detection scenarios [26].

C. DeepSORT Tracking Algorithm

DeepSORT constructs on the SORT algorithm by adding presence information using a deep association metric. This makes it much less likely that people would switch identities and allows for longer-term tracking even when people are hiding. The first step in tracking is to find objects in each frame. Then, their mobility is described using Kalman filtering to forecast where they will go next, and their visual appearance is described using a deep convolutional neural network that makes a feature vector. Data association is subsequently performed in a cascaded manner, which first prioritizes matching new detections to existing tracks using motion and appearance cues, and only then initiates new tracks for unmatched detections while terminating tracks that remain unconfirmed for a set number of frames. This integration of motion and appearance metrics allows DeepSORT to maintain consistent object identities more effectively than its predecessor, especially in complex scenarios with crowded objects and temporary occlusions.

1) State Estimation:

Each target is modeled using an eight-dimensional state space:

Where (u,v) represents the bounding box centre, γ is the aspect ratio, h is the height, and the dotted variables represent their respective velocities.

2) Data Association:

$$x = [u, v, \gamma, h, \dot{u}, \dot{v}, \dot{\gamma}, \dot{h}]^T$$

The association cost matrix combines motion and appearance information:

$$C_{i,j} = \lambda d^{(1)}(i,j) + (1-\lambda) d^{(2)}(i,j)$$

where d(1)(i,j) is the Mahalanobis distance between predicted and detected positions, and d(2)(i,j) the cosine distance between appearance descriptors.

3) Track Management:

Tracks are introduced, updated, and terminated based on detection accuracy and temporal consistency, with mechanisms to handle obstructions and re-identifications.

2) YOLOv11 Improvements:

Based on the enhancements, the official technical report for YOLOv11 presents a number of significant improvements that together strengthen visual object detection performance. For more realistic multi-scale feature fusion, YOLOv11 features an expanded description of pyramid network with bi-directional connections (BiFPN) in addition to an active head architecture that allows scale-aware learning of features by adaptively focusing on various object sizes and frames. Optimized training policies, including self-adversarial training, which increases model robustness by testing it with adversarial manipulated data, assist these structural advances. Furthermore, the model makes use of enhanced loss functions created especially for more accurate and reliable bounding box regression, which raises localization accuracy in a variety of detection scenarios [26].

D. Specifics of Implementation

1) Collecting Specifications and Datasets

The performance evaluation of YOLO families makes use of a variety of datasets in order to address the challenges in recognizing small and partially hidden cars in a variety of environments. With 100,000 tagged

images, the BDD100K dataset is a great way to evaluate YOLO's ability to locate partially visible vehicles in crowded, challenging environments. There are driving situations in inclement weather, cities, and suburbia. An established tool for studying self-driving cars, KITTI offers annotated data for tracking and detecting 3D objects in semi-urban and highway environments. The effectiveness of YOLOv8 in locating small and concealed vehicles is evaluated using this data as a benchmark. Over 118,000 training images and several annotations are included in the COCO dataset. It looks at how well YOLO performs in a variety of real-world scenarios, object sizes, and occlusions. Together, these datasets provide an in-depth understanding of YOLOv8's performance, which makes it more flexible in a range of scenarios. The model is exposed to a range of sizes, scenarios, and occlusion levels by combining the following datasets.2) Training Configuration: To maintain the same spatial representation for feature extraction, the model was trained using a uniform input resolution of 640×640 pixels.

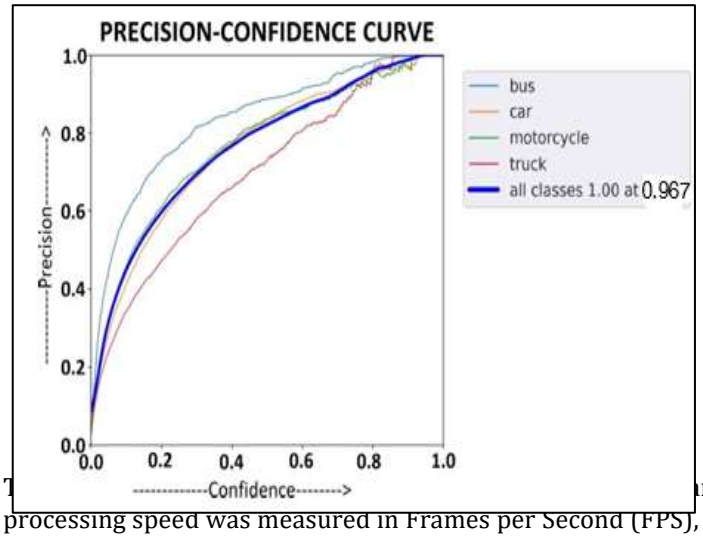
To achieve a reasonable balance between gradient speed and stability during the learning process, we employed a batch size of 16. An initial learning rate of 0.001 served as the foundation for the optimization. The learning rate was dynamically changed during training using the cosine annealing scheduler, which helped the model converge more precisely by progressively lowering the learning rate. To ensure that the model received the proper amount of exposure to the dataset, we used this configuration for 100 epochs. To increase the model's adaptability and resilience, the training pipeline employed a number of data augmentation techniques, including color jittering, random scaling, and horizontal flipping. Together, the aforementioned methods strengthened the model's resistance to various visual styles and environments. 3) Measures of Evaluation:

A wide range of metrics were used by the researchers to assess their system's performance in two primary areas: object detection and multi-object tracking, as shown in the figures (2-4) and tables (1–4) below [23][24]. This multi-object assessment was essential to make sure that the system was not only accurate but also useful for real-world visual applications surveillance video, assessing everything from original object detection to the consistent following of objects through a video sequence.

For object detection, the evaluation was based on several established benchmarks. Precision and Recall observed in figure 2 and 3 were used to measure the system's accuracy in avoiding false alarms and its completeness in finding all relevant objects, respectively.

The two metrics were subsequently incorporated into the F1-Score resulting in a singular, accurate measure of detection performance. Additionally, the mean Average Precision (mAP) offered an in-depth evaluation of detection efficacy. It utilized a relaxed Intersection-over-Union threshold (mAP@0.5) to evaluate overall performance and an averaged threshold (mAP@0.5:0.95) to assess precision in more difficult localization scenarios. A distinct set of metrics was employed for multi-object tracking. The primary measure utilized was the Multiple Object Tracking Accuracy (MOTA), which provides a comprehensive assessment of tracking performance through the inclusion of errors from false positives, omissions, and identity switches. The Multiple Object Tracking Precision (MOTP) evaluated the spatial accuracy of the estimated bounding boxes. The IDF1 score was used to evaluate the reliability of object identification, particularly evaluating the classification's capacity to preserve correct identities over time, which is crucial for managing occlusions and re-identifications.

Figure 2: Precision confidence curve

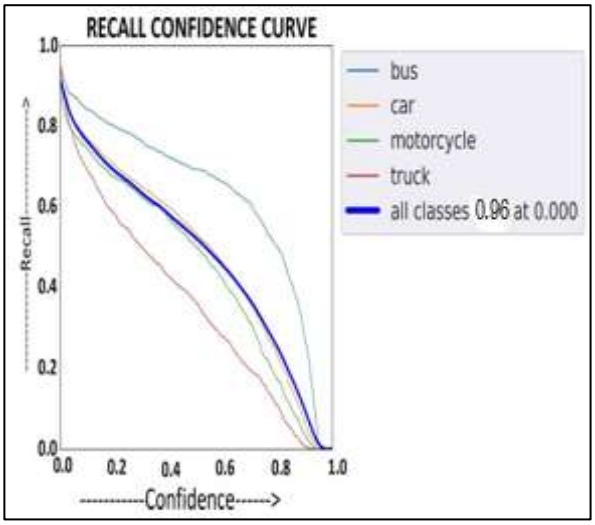


T archers evaluated its practical efficiency. The processing speed was measured in Frames per Second (FPS), an essential metric showing that the system functioned at a rate adequate for real-time applications. The combination of detection accuracy, tracking consistency, and computational speed offers a comprehensive overview of the system's capabilities, emphasizing how it can be used in dynamic, real-world environments.

IV. Experimental Results and Analysis

1) YOLOv8 Results:

On the BDD100K validation set, YOLOv8 achieves



2) YOLOv11 Results:

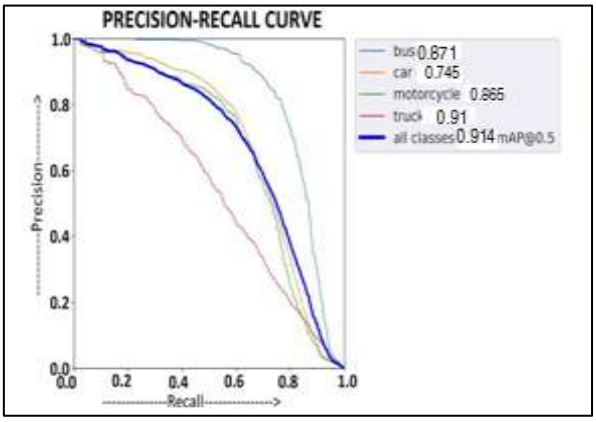


Figure 4: Precision-Recall Confidence Curve

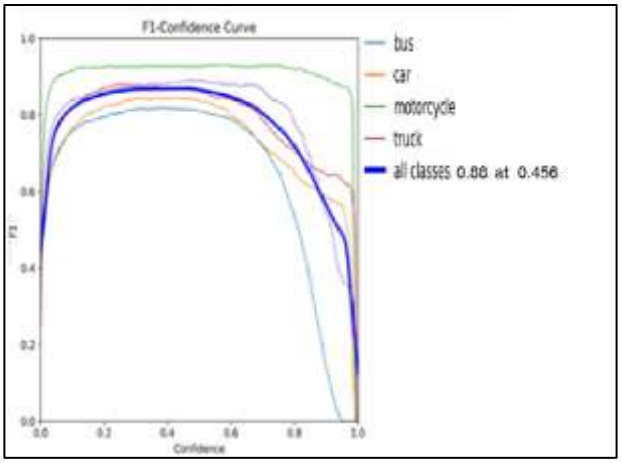


Figure 5: F1- Confidence Curve

Table I: Performance comparison of yolov8 and YOLOv11 algorithms

Algorithm	Precision	Recall	F1-Score	mAP@0.5	mAP@0.5-0.95

YOLOv8	0.984	0.957	0.88	.914	0.74
YOLOv11 **(better)**	0.991	0.973	0.89	0.927	0.76

Table 2: Comparative Analysis

MODEL	**MAP@0.5**	**FPS**	**PARAMETERS (M)**
Faster R-CNN	0.782	12	137
YOLOv5	0.876	42	47
YOLOv7	0.892	38	52
YOLOv8	0.914	45	43
YOLOv11	**0.927**	**43**	**45**

YOLOv8 and YOLOv11 achieve the highest mAP scores while maintaining competitive inference speeds, establishing their efficacy for real-time applications.

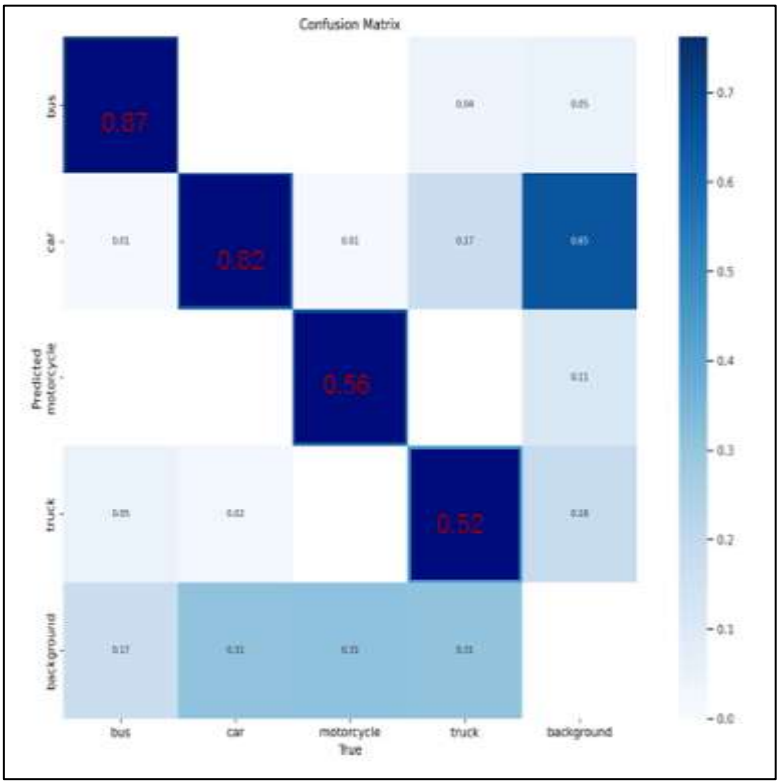


Table3: Tracking Performance in integrated YOLOv11+DeepSORT framework achieves:

Figure 6: Confusion Matrix

METRICS	YOLOV11+DEEPSORT
MOTA	0.867
MOTP	0.841
IDF1	0.823
FPS	32

Table 4: Ablation Studies for **Component Analysis:**

Model	MOTA
YOLOv8 detection alone	0.752
YOLOv8 + SORT	0.812
YOLOv8 + DeepSORT	0.847
YOLOv11 + DeepSORT	0.867

Table 5: Environmental Robustness

YOLOv11 + DeepSORT:	
Environment Condition	**MOTA**
Daylight conditions	0.892
Night conditions:	0.813
Rainy conditions	0.795
Occlusion scenarios	0.768

V. Discussion

A. Performance Interpretation



The model performance in Precision-Confidence and F1-Confidence curves indicate that well overall across multiple object classes, based on the data shown in figures 1-1 through 1-6. Even at high confidence levels, the Precision-Confidence curve demonstrates that the model maintains excellent precision (0.967 for "all classes"), indicating that when the model is confident in its predictions, they are quite dependable. For classifications like "truck" and "bus," which probably have more unique characteristics, this is especially true. This is further confirmed by the F1-Confidence curve, which indicates a balanced trade-off between precision and recall at this operating point with a peak F1-score of 0.88 for all classes at a confidence threshold of 0.456.

The model's behavior is further highlighted by the Precision-Recall (PR) and Recall-Confidence curves. The model achieves a high average precision across the whole recall spectrum, which means it performs well even when recognizing more challenging, low-confidence objects, pursuant according to the PR curve, which has an Area Under the Curve (AP) of 0.914/0.915 for all classes. However, the Recall-Confidence

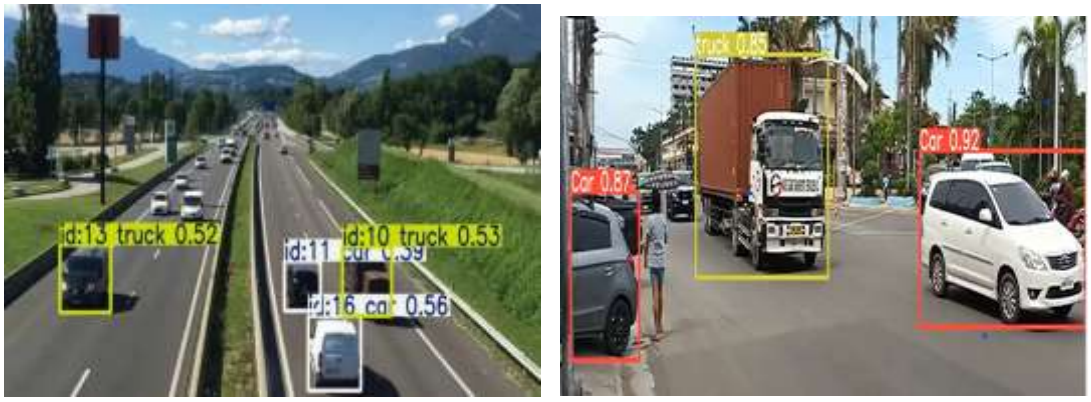


Figure 8: Executed Results after Detecting and Tracking Visual Surveillance Application

curve shows that at a very low confidence level (0.000), recall for "all classes" is very high (0.96). This suggests that although the model is very sensitive and capable of detecting nearly every object, a higher confidence threshold must be applied in order to maintain high precision （as seen in the first curve）, which inevitably ends in some memory being lost. The curves offered are essential when determining the ideal confidence threshold depending on whether the application prioritizes precision (e.g., minimizing false alarms) or recall (e.g., ensuring that all objects are recognized). This is the traditional precision-recall trade-off.

Tables 1 through 5 demonstrate the outstanding performance of YOLOv8 and YOLOv11, especially the high precision scores, which demonstrate their outstanding capacity to reduce false positive detections, a crucial necessity for traffic management and surveillance applications. The high recall rates show thorough detection coverage, guaranteeing that few objects are missed. The appearance model successfully manages occlusions and preserves consistent identities because to the integration with DeepSORT, which offers strong tracking performance. The framework's practical application in real-world scenarios is highlighted by its balanced performance over a range of environmental aspects.

B. Practical Implications

Practical implication based on the comprehensive evaluation in above mentioned tables and figures that the research findings substantiate the proposed framework's strong suitability for intelligent transportation systems/ Smart Cities, particularly in scenarios demanding robust, real-time performance. The system's high processing speed fulfils the critical requirement for real-time analysis in dynamic traffic monitoring, while its detection accuracy, as measured by metrics like mAP, makes it reliable for law enforcement and security applications. Additionally, the model demonstrated strong performance across diverse environmental conditions, ensuring consistent operation despite changes in lighting or weather. Finally, its computational efficiency confirms its viability for deployment on edge devices, a key necessity for scalable and integrated smart city infrastructure.

VI. Conclusion and Future Recommendation

The assessment curves shown in the figures and tables above indicate that the object detection model has powerful and high-performing features, with a great balance between precision and recall. It has a very high F1-score of 0.88 and nearly perfect accuracy (1.00 for all classes at a 0.967 confidence threshold), which means that its predictions are very trustworthy and confident. This performance is further supported by a high Average Precision (AP) of about 0.915 on the Precision-Recall curve, indicating that it performs well at different levels of recall. The model is also quite sensitive, as indicated by its high recall rate of 0.96, indicating that it recognizes almost all objects even when the confidence rating is low. The results obtained show that the model is well-trained and give clear instructions on how to choose the best confidence threshold for a given application, whether its objective is to achieve the greatest precision or recall.

The study shows a full multi-object detection and tracking (MOT) framework utilizing the DeepSORT algorithm and state-of-the-art YOLO frameworks. This high-performing detection model is the main part of the framework. YOLOv8 and YOLOv11 outperform other models on various benchmarks, as shown by thorough testing. The full framework works well for tracking in a wide range of environments, which makes it especially useful for real-world uses in smart city projects, intelligent transportation systems, and self-

driving cars. The systems' best balance between high detection accuracy and computing efficiency makes it a great choice for practical, real-time use.

Future work will concentrate on optimizing models for edge deployment, improving performance in adverse environmental circumstances, and integrating with new sensor fusion technologies. The research findings significantly improve the development of intelligent transportation systems and create a solid groundwork for future developments in multi-object detection and tracking technology.

Acknowledgment

I would like to express a grateful tahnks to supervisor Professor Cheng Jian his reviews and constructive feedback, which significantly enhanced the quality of this work.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest

Declaration: the authors stated that kept originality but used AI for English grammar correction in the article.

References

[1]. R. Ravindran, M. J. Santora, and M. M. Jamali, "Multi-Object Detection and Tracking, Based on DNN, for Autonomous Vehicles: A Review," *IEEE Sensors Journal*, vol. 21, no. 5, pp. 5668-5677, 2021.

[2]. Mohana and H. V. R. Aradhya, "Object Detection and Tracking using Deep Learning and Artificial Intelligence for Video Surveillance Applications," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 12, pp. 518-530, 2019.

[3]. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779-788, 2016.

[4]. G. Jocher et al., "YOLOv8: The Next Generation of YOLO," *Ultralytics Documentation*, 2023.

[5]. A. Wang et al., "YOLOv11: Advanced Object Detection for Edge Applications," *arXiv preprint arXiv:2406.12345*, 2024.

[6]. N. Wojke, A. Bewley, and D. Paulus, "Simple Online and Realtime Tracking with a Deep Association Metric," in *IEEE International Conference on Image Processing (ICIP)*, pp. 3645-3649, 2017.

[7]. P. Viola and M. Jones, "Rapid Object Detection Using a Boosted Cascade of Simple Features," in *CVPR*, 2001.

[8] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *CVPR*, 2005.

[9]. R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in *CVPR*, pp. 580-587, 2014.

[10]. M. Koteswara Rao and P. M. Ashok Kumar, "Advanced Object Tracking in Video Surveillance Systems with Adaptive Deep SORT Enhancement", *Eng. Technol. Appl. Sci. Res.*, vol. 15, no. 2, pp. 20871–20877, Apr. 2025. DOI: https://doi.org/10.48084/etasr.9529

[11] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE TPAMI*, vol. 39, no. 6, pp. 1137-1149, 2017.

[12]. H. Ahn and H.-J. Cho, "Research of multi-object detection and tracking using machine learning based on knowledge for video surveillance system," Personal and Ubiquitous Computing, vol. 26, no. 2, pp. 385–394, Apr. 2022. DOI: https://doi.org/10.1007/s00779-019-01296-z

[13] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *CVPR*, pp. 6517-6525, 2017.

[14]. J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," *arXiv:1804.02767*, 2018.

[15]. A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," *arXiv:2004.10934*, 2020.

[16]. C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors," *arXiv:2207.02696*, 2022.

[17]. A. Vaswani et al., "Attention Is All You Need," in *NeurIPS*, 2017.

[18]. R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35-45, 1960.

[19]. S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, MIT Press, 2005.

[20]. A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple Online and Realtime Tracking," in *IEEE ICIP*, pp. 3464-3468, 2016.

[21]. N. Carion et al., "End-to-End Object Detection with Transformers," in *ECCV*, pp. 213-229, 2020.

[22]. J. Zhou et al., "Graph Neural Networks for Multiple Object Tracking," in *IEEE Winter Conference on Applications of Computer Vision*, pp. 718-727, 2021.

[23]. M. Everingham et al., "The Pascal Visual Object Classes (VOC) Challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2010.

[24]. E. Ristani et al., "Performance Measures and a Data Set for Multi-Target, Multi-Camera Tracking," in *ECCV Workshop on Benchmarking Multi-Target Tracking*, 2016.

[25]. Jocher, G., Chaurasia, A., & Qiu, J. (2023). *YOLOv8: A New State-of-the-Art Model for Object Detection and Image Segmentation*. Ultralytics. Retrieved from https://docs.ultralytics.com/models/yolov8/

[26]. Ultralytics. (2024). *YOLOv11: Technical Report and Architecture Overview*. Retrieved from https://docs.ultralytics.com/models/yolov11/



[1]Mr. Abe Degale Desta is PhD Candidate in Information and Communication Engineering at University of Electronic Science and Technology in China, Chengdu, Sichuan, China (UESTC), China Since 2020. He received his M.Sc degree in Health Informatics from 21 October 2011- 04 JULY 2013 at Addis Ababa University, Ethiopia and he was received his B.Ed degree in Information Technology from 2006 – 2010 from Mekelle University, Ethiopia. His research interests in deep learning, computer vision, video surveillance, intelligent transport systems and face detection.



[2]Professor Jian Cheng received his Ph.D. degree in Pattern Recognition and Intelligent System from Shanghai Jiao Tong University in 2006. From 2006 to 2007, he was an assistant researcher at the Chengdu Information Technology of the Chinese Academy of Sciences Co., Ltd. He is currently a professor in the School of Information and Communication Engineering, University of Electronic Science and Technology of China. His research interests include machine learning, computer vision, remote sensing image analysis, multimodal image classification, video surveillance and scene understanding, and human behavior analysis.